

# 月面探査ローバーのオペレーション 開発におけるROSとGazeboの活用

The background image shows a simulated lunar surface with a grey, cratered terrain. In the center-right, a lunar lander with four legs is positioned. In the bottom right corner, a small rover with four yellow wheels and a vertical mast is visible. The scene is lit from the left, creating shadows on the right.

Fabian Dubois, Engineer, ispace inc.

# 自己紹介・ファビアン デュボア

フランス出身、2009年Supelec卒業

2007年九州大学:ロボットのステレオ画像を活用した位置測定

2009年ZMP:ソフト・電子設計開発

2009年Orange:ロボット/IoTを活用したサービス開発

2013年Locarise共同設立:IoTを活用した、屋内位置情報サービス開発

2015年～Datamaplab設立:IoTとAIを活用したシステム開発

2017年～Hakuto/ ispace inc.:ローバーのオペレーション、シミュレーション、アルゴリズム開発



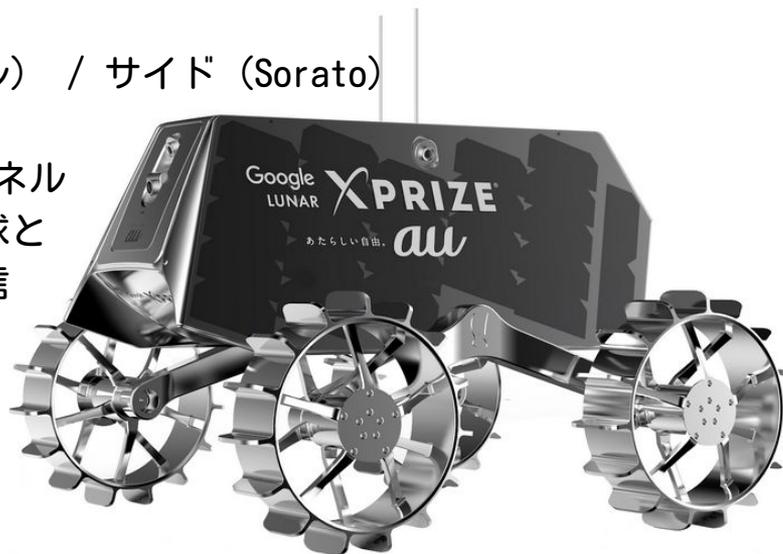
# コンテンツ

1. ロバー・ミッション紹介
2. 月面着陸したローバーの操作の制約
3. Gazeboでのテスト環境作成
4. GazeboとROSの活用

# ispaceの月面探査ローバー「Sorato」

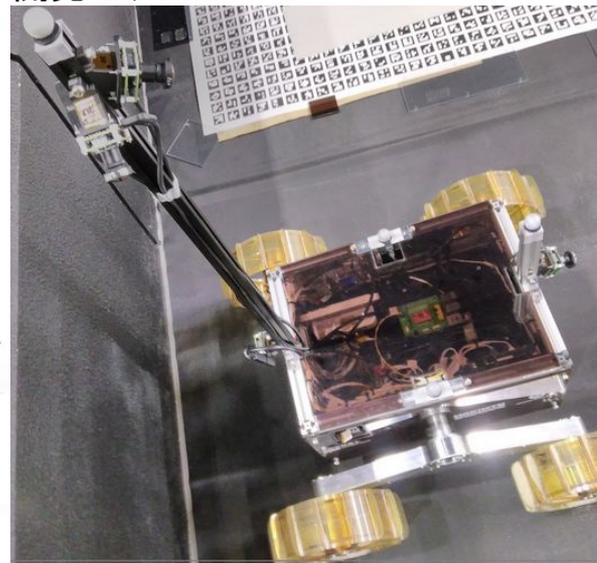
## スペック

- 4輪、スキッドステア、パッシブサスペンション  
10cm/s
- カメラ:前面、背面  
+ マスト(開発モデル) / サイド (Sorato)
- 深度センサー: T0F
- 電池 + 太陽光発電パネル
- ランダーを通じた地球と  
ローバー間の中継通信
- 50cm/4kg
- 10cm/s



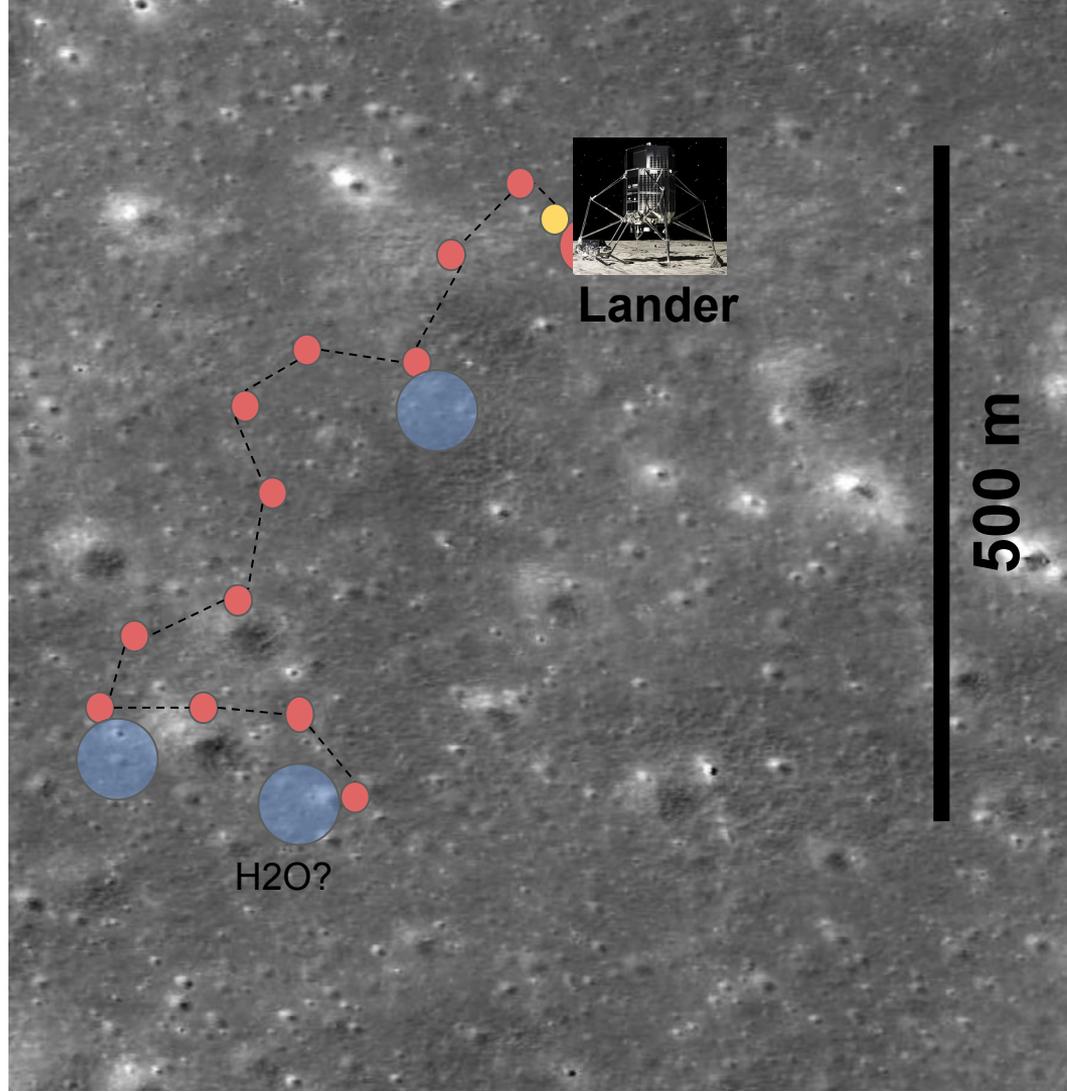
Sorato

## 開発モデル



# HAKUTO-R ミッション

- 搭載センサーで月面測定計画を実施
- 予定走行距離1~10キロ
- H2O



# 月面着陸したローバーの操作の制約

# 制約1: 14日間以下のミッション

- 太陽光発電なので太陽が出ていないと走行できない
- 短い! 試運転の時間も少ない。
- 激しい温度変動 (-170~120°C)

## 制約2: 通信



The diagram shows a full moon at the top center against a black background. Below it, a blue gradient represents the Earth's atmosphere and surface. A red dashed line forms a loop between the moon and the Earth's surface. The left side of the loop is labeled '1.25s uplink' and the right side is labeled '1.25s downlink'. Red arrows at the ends of the dashed line indicate the direction of communication: one arrow points up towards the moon, and another points down towards the Earth's surface.

1.25s  
uplink

1.25s  
downlink

- 通信遅延
- データレートも限られている

# 制約3: GPSも方位磁針も使えない

ランダー相対ナビゲーションオドメトリー

太陽によるナビゲーション

=>カメラでローバー自身の位置、向きを把握します。

Intrepid  
Descent  
Stage

Crater

Surveyor  
Crater

Surveyor 3  
Spacecraft

Bench  
Crater

Sharp  
Crater

Apollo 12 seen by LRO NASA's Goddard Space Flight Center/ASU

Intrepid 5X Enlargement



## 制約4： 困難な照明条件

- 大気がない
- 光が強くて露出過度になりやすい
- 影が真っ黒に映る
- 地面の詳細を把握しづらい。



# 未知数な環境

- どうやってアルゴリズムの動作を確認する？
- どうやってオペレーションプランの確認、最適化をする？
- どうやってオペレーターのトレーニングを行う？

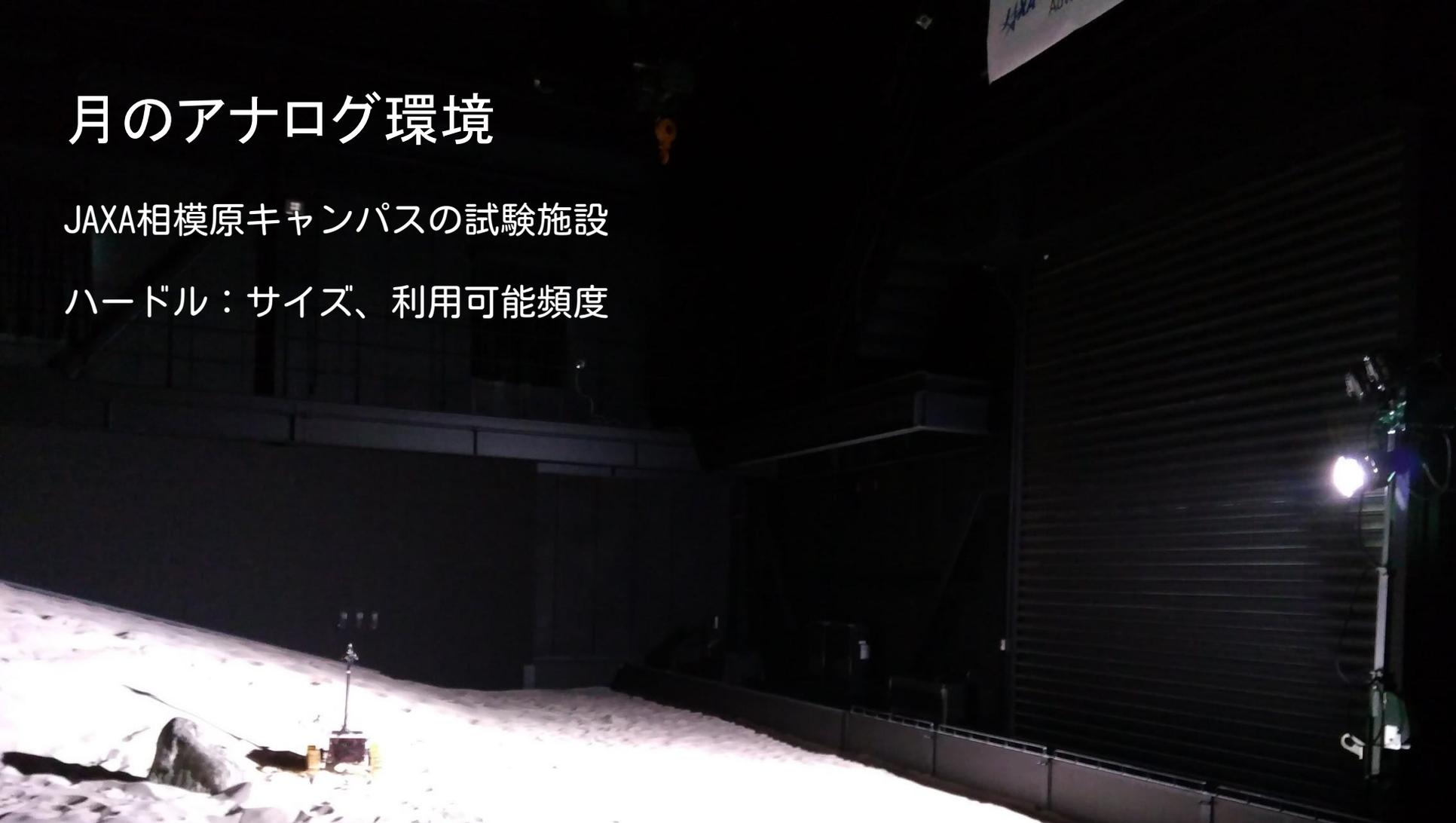
地球上に月を作ろう



# 月のアナログ環境

JAXA相模原キャンパスの試験施設

ハードル：サイズ、利用可能頻度



A simulated lunar surface with a lunar lander and a rover. The lander is a black cylindrical structure with four legs, standing on a grey, cratered surface. A small rover is visible to the left. The background is a vast, cratered landscape under a dark sky.

# Gazeboの登場！

Gazebo9を使用

物理エンジンODE、3次元レンダリングOgre

物理精度より、カメラのシミュレーションの方が重要

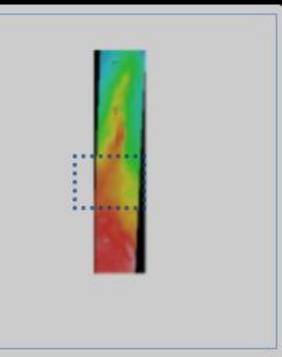


# Reference DEM(デジタル標高モデル)

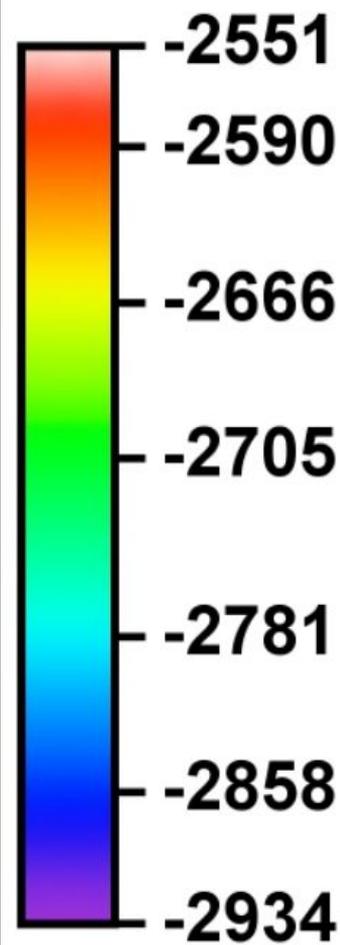
NASA LRO (Lunar Reconnaissance Orbiter) から, 5m/px

[http://wms.lroc.asu.edu/lroc/view\\_rdr/NAC\\_DTM\\_CHANGE3](http://wms.lroc.asu.edu/lroc/view_rdr/NAC_DTM_CHANGE3)

100x100m 地形 -> 20x20px



Elevation (m)

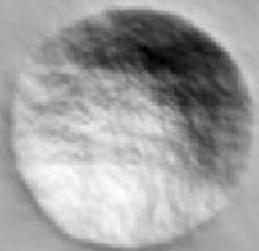


# 地形を実物に近づけるためにクレーターを作成

観測されたサイズ分布を利用してpythonでクレーターを作成.

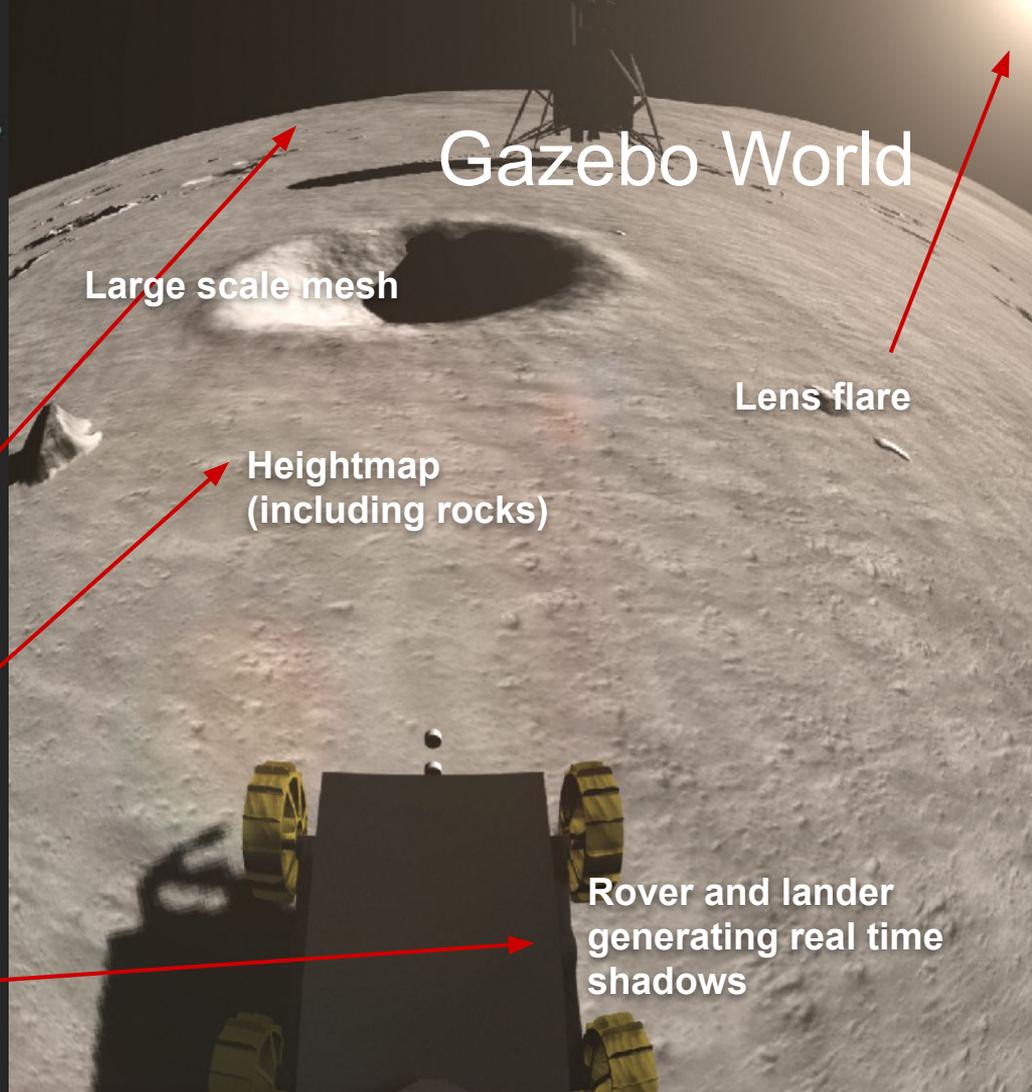
5m/pxからto 2cm/px. (物理エンジン設定に注意)

4096x4096のDEM→Level Of Details plugin使用.



```
<light name='sun' type='directional'>
  <cast_shadows>1</cast_shadows>
  <pose frame=''>0 0 0 0 1.02 -1.57</pose>
  <diffuse>1 1 1 1</diffuse>
  <specular>0.8 0.8 0.8 1</specular>
  <attenuation>
    <range>1000</range>
    <constant>1</constant>
    <linear>0</linear>
    <quadratic>0</quadratic>
  </attenuation>
</light>
```

```
<include>
  <uri>model://moon_dem</uri>
  <pose>0 0 5 0 0 0</pose>
</include>
<include>
  <uri>model://dem_2k</uri>
  <pose>0 0 -1.04 0 0 0</pose>
</include>
<include>
  <uri>model://devrover</uri>
  <pose>0 0 0.8 0 0 0</pose>
</include>
```



# High resolution heightmapのモデル

```
1 <?xml version="1.0"?>
2 <sdf version="1.6">
3   <model name="moon_dem">
4     <static>1</static>
5     <link name='link'>
6       <collision name='collision'>
7         <geometry>
8           <heightmap>
9             <uri>model://moon_dem/materials/textures/dem.tif</uri>
10            <size>40 40 0.4</size>
11            <pose>0 0 -1</pose>
12          </heightmap>
13        </geometry>
14      </collision>
15      <visual name='visual'>
16        <cast_shadows>0</cast_shadows>
17        <geometry>
18          <heightmap>
19            <use_terrain_paging>>false</use_terrain_paging>
20            <uri>model://moon_dem/materials/textures/dem.tif</uri>
21            <size>40 40 0.4</size>
22            <pose>0 0 -1</pose>
23          </heightmap>
24        </geometry>
25        <material>
26          <script>
27            <uri>model://moon_dem/materials/scripts/heightmap_custom.material</uri>
28            <name>Gazebo/CustomHeightmapShader</name>
29          </script>
30        </material>
31        <plugin name="lod" filename="libHeightmapLODPlugin.so">
32          <lod>6</lod>
33          <skirt_length>0.5</skirt_length>
34        </plugin>
35      </visual>
36    </link>
37  </model>
38 </sdf>
```

```
1 <?xml version="1.0"?>
2 <sdf version="1.0">
3   <model name="moon_demo">
4     <static true />
5     <link name="moon_demo">
6       <collision>
7         <geometry>
8           <heightmap>
9             <plugin name="lod" filename="libHeightmapLODPlugin.so">
10              <lod>6</lod>
11              <skirt_length>0.5</skirt_length>
12            </plugin>
13          </heightmap>
14        </geometry>
15      </collision>
16      <visual name="visual">
17        <cast_shadows>0</cast_shadows>
18        <geometry>
19          <heightmap>
20            <use_terrain_paging>false</use_terrain_paging>
21            <uri>model://moon_demo/materials/textures/dem.tif</uri>
22            <size>40 40 0.4</size>
23            <pose>0 0 -1</pose>
24          </heightmap>
25        </geometry>
26        <material>
27          <script>
28            <uri>model://moon_demo/materials/scripts/heightmap_custom.material</uri>
29            <name>Gazebo/CustomHeightmapShader</name>
30          </script>
31        </material>
32        <plugin name="lod" filename="libHeightmapLODPlugin.so">
33          <lod>6</lod>
34          <skirt_length>0.5</skirt_length>
35        </plugin>
36      </visual>
37    </link>
38  </model>
39 </sdf>
```

Level of Details Plugin

```
<?xml version="1.0"?>
```

```
<material>
```

```
  <script>
```

```
    <uri>model://moon_dem/materials/scripts/heightmap_custom.mate
```

```
    <name>Gazebo/CustomHeightmapShader</name>
```

```
  </script>
```

```
</material>
```

```
<visual name='visual'>
```

```
  <cast_shadows>0</cast_shadows>
```

```
  <geometry>
```

```
    <heightmap>
```

```
      <use_terrain_paging>>false</use_terrain_paging>
```

```
      <uri>model://moon_dem/materials/textures/dem.tif</uri>
```

```
      <size>40 40 0.4</size>
```

```
      <pose>0 0 -1</pose>
```

```
    </heightmap>
```

```
  </geometry>
```

```
  <material>
```

```
    <script>
```

```
      <uri>model://moon_dem/materials/scripts/heightmap_custom.material</uri>
```

```
      <name>Gazebo/CustomHeightmapShader</name>
```

```
    </script>
```

```
  </material>
```

```
  <plugin name="lod" filename="libHeightmapLODPlugin.so">
```

```
    <lod>6</lod>
```

```
    <skirt_length>0.5</skirt_length>
```

```
  </plugin>
```

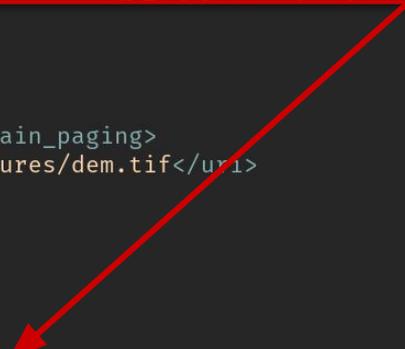
```
</visual>
```

```
</link>
```

```
</model>
```

```
</sdf>
```

Custom shaders



# カスタムなシェーダーマテリアル

gazeboのレポジトリにある例を参考にできます:

[https://bitbucket.org/osrf/gazebo/src/default/test/media/materials/scripts/heightmap\\_custom\\_fp.glsl](https://bitbucket.org/osrf/gazebo/src/default/test/media/materials/scripts/heightmap_custom_fp.glsl)

```
50 material Gazebo/CustomHeightmapShader
51 {
52     technique
53     {
54         pass
55         {
56             vertex_program_ref Gazebo/HeightmapVS
57             {
58             }
59             fragment_program_ref Gazebo/HeightmapFS
60             {
61             }
62             texture_unit shadowMap0
63             {
64                 content_type shadow
65                 tex_address_mode clamp
66             }
67             texture_unit shadowMap1
68             {
69                 content_type shadow
70                 tex_address_mode clamp
71             }
72             texture_unit shadowMap2
73             {
74                 content_type shadow
75                 tex_address_mode clamp
76             }
77             texture_unit NormalTexture
78             {
79                 texture sand_normal.png 2d
80                 filtering anisotropic anisotropic linear
81                 max_anisotropy 16
82                 tex_address_mode clamp
83             }
84             texture_unit diffuseMap
85             {
86                 texture diffuse_shadows_4096_1000.jpg 2d
87                 filtering anisotropic anisotropic linear
88                 max_anisotropy 16
89                 tex_address_mode clamp
90             }
91             texture_unit NormalMap
92             {
93                 texture dem_normal_4096_1000.png 2d
94                 filtering anisotropic anisotropic linear
95                 max_anisotropy 16
96                 tex_address_mode clamp
97             }
98         }
99     }
100 }
```

# Custom shaders

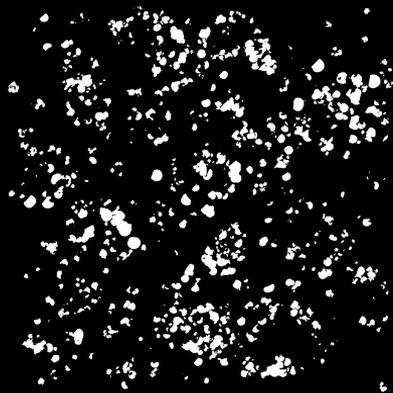
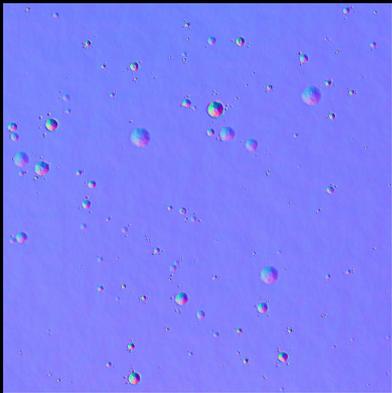
OpenGL glsl vertex and  
fragment shader programs

```
50 material Gazebo/CustomHeightmapShader
51 {
52     technique
53     {
54         pass
55         {
56             vertex_program_ref Gazebo/HeightmapVS
57             {
58             }
59             fragment_program_ref Gazebo/HeightmapFS
60             {
61             }
62         }
63     }
64     texture_unit shadowMap0
65     {
66         content_type shadow
67         tex_address_mode clamp
68     }
69     texture_unit shadowMap1
70     {
71         content_type shadow
72         tex_address_mode clamp
73     }
74     texture_unit shadowMap2
75     {
76         content_type shadow
77         tex_address_mode clamp
78     }
79 }
```

# Custom shaders

Pass needed texture files.

Normal blending .. multi textures.. From terrain logic.



```
74     content_type shadow
75     tex_address_mode clamp
76 }
77 texture_unit NormalTexture
78 {
79     texture sand_normal.png 2d
80     filtering anisotropic anisotropic linear
81     max_anisotropy 16
82     tex_address_mode clamp
83 }
84 texture_unit diffuseMap
85 {
86     texture diffuse_shadows_4096_1000.jpg 2d
87     filtering anisotropic anisotropic linear
88     max_anisotropy 16
89     tex_address_mode clamp
90 }
91 texture_unit NormalMap
92 {
93     texture dem_normal_4096_1000.png 2d
94     filtering anisotropic anisotropic linear
95     max_anisotropy 16
96     tex_address_mode clamp
97 }
98 }
99 }
100 }
```

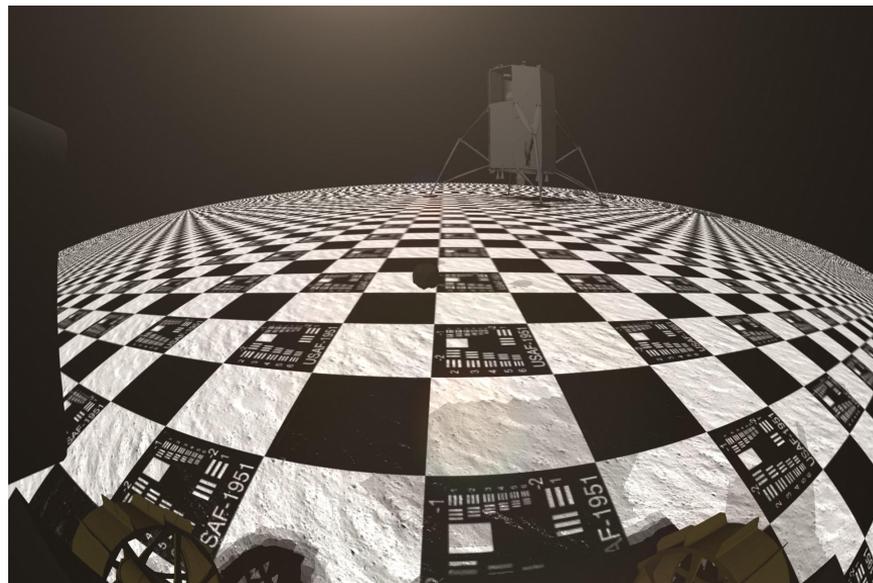
# GazeboとROSを活用したローバーの操作



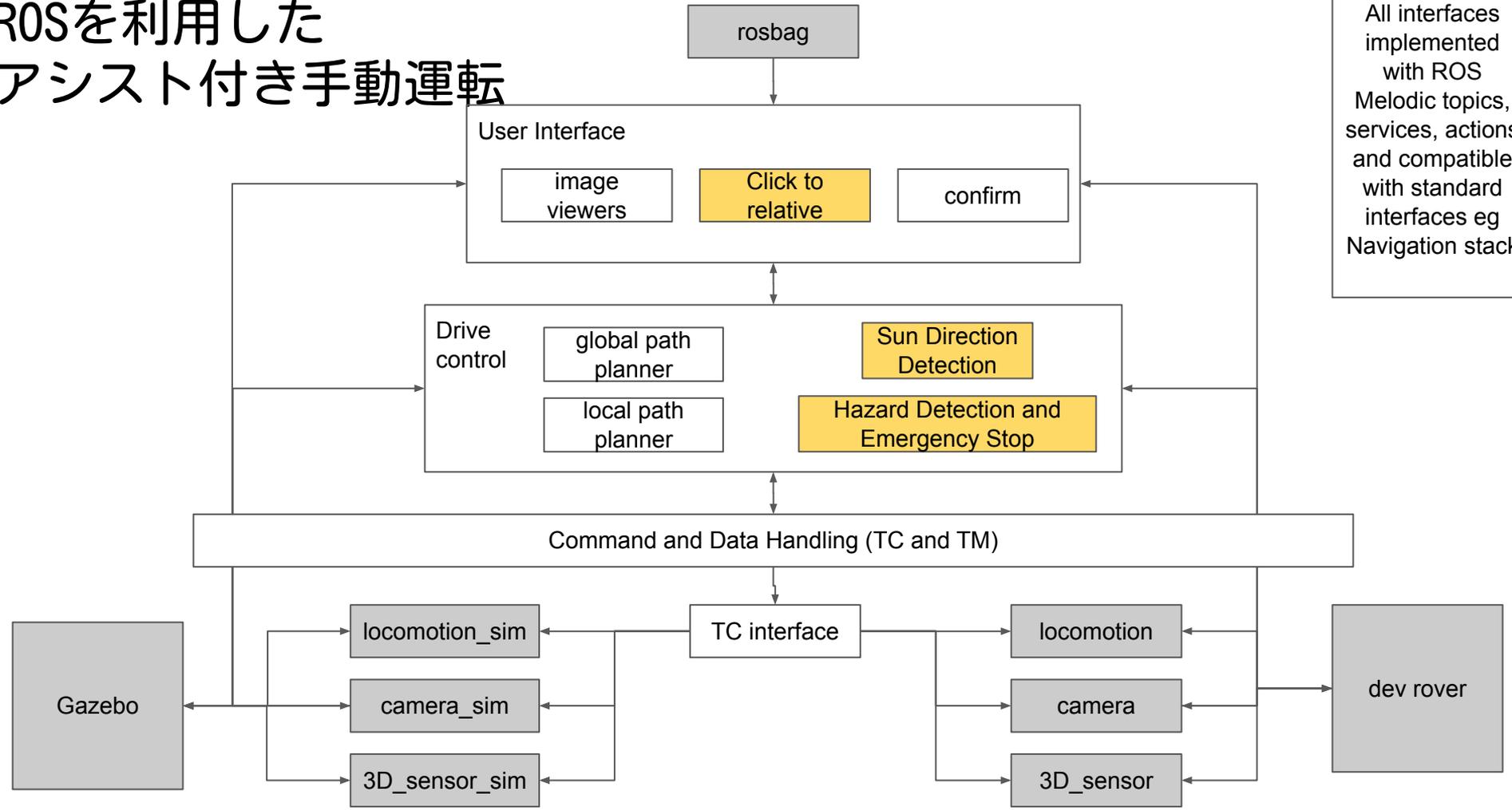


Lunokhod

# Gazeboを利用してカメラの配置を最適化します



# ROSを利用した アシスト付き手動運転



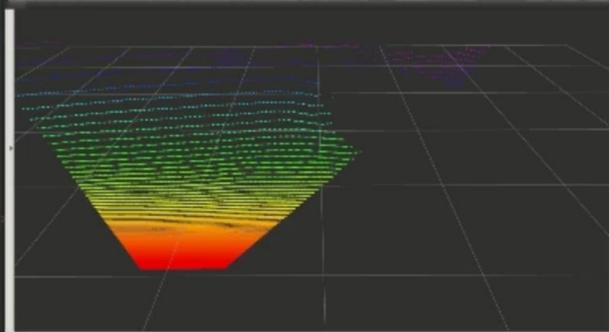
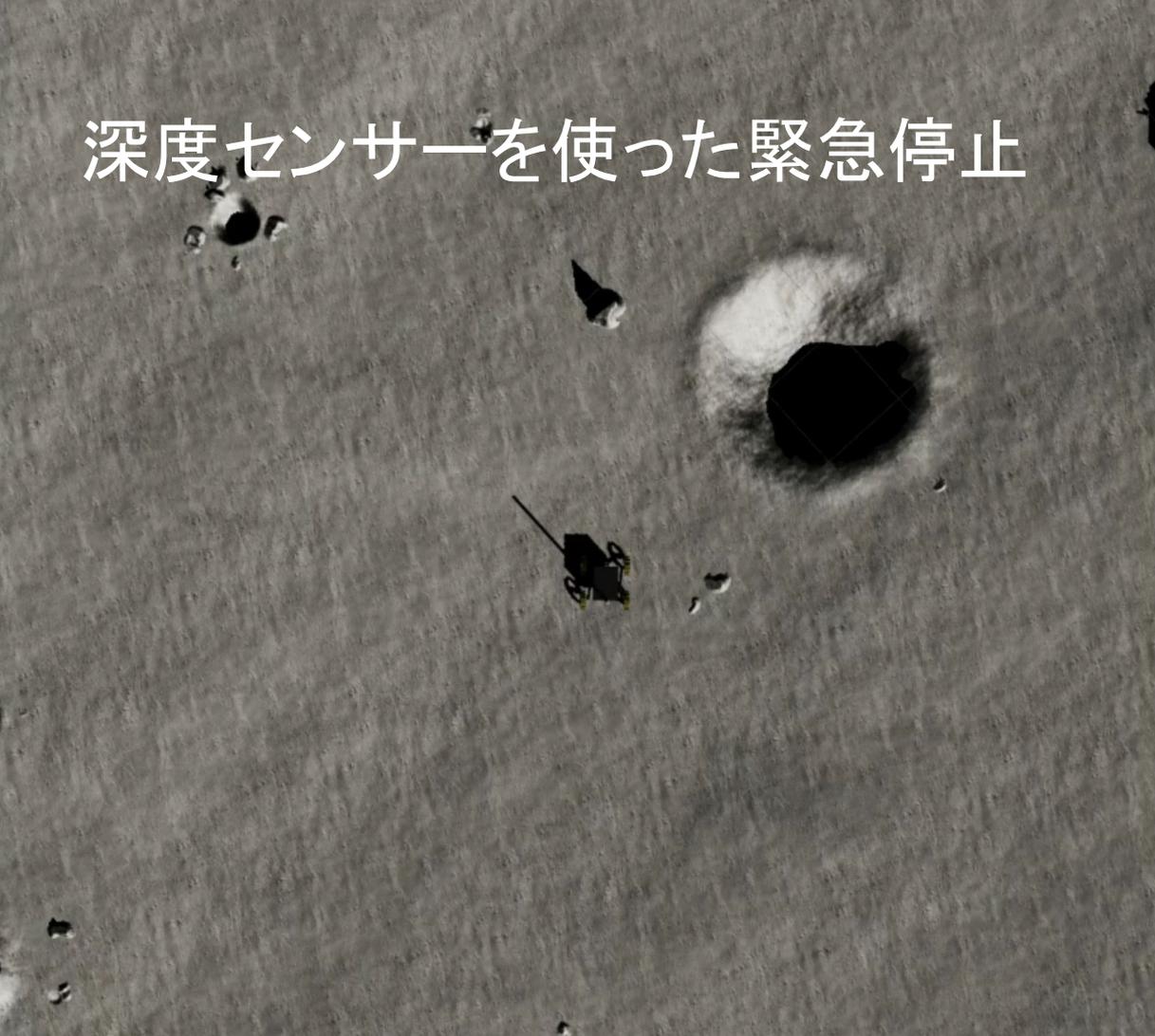
All interfaces implemented with ROS Melodic topics, services, actions and compatible with standard interfaces eg Navigation stack

# 「Click and go」操作

- spot turn
- 直進



# 深度センサーを使った緊急停止



Time  
ROS Time: 69.42 ROS Elapsed: 60.38 Job Time: 1569231051.10 Total Elapsed: 50:20  Covertorial  
Keys: Left-Click: Rotate, Middle-Click: Move X/Y, Right-Click/Mouse Wheel: Zoom, Shift: Move camera. 31 FPS

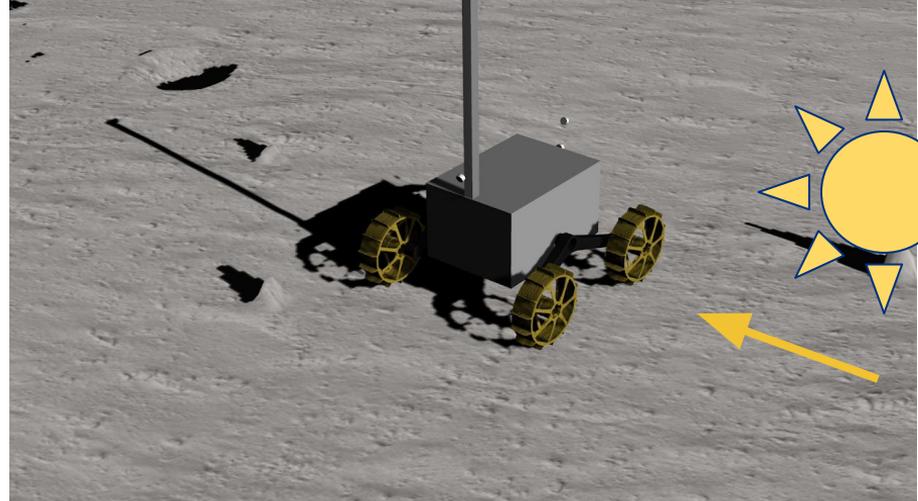
# 太陽による方向認識

機械学習で太陽の相対位置測定を行いました。

Gazeboを使って、数千アイテムのデータセットを作成：

- 実際の姿勢データ
- カメラ画像

analogue環境に集めたデータで再確認できました。



# 今後の計画

- Gazebo Ignitionへの移行を楽しみにしている (複数のカメラにはまだ対応していない) => Physically based rendering, OptiX(ray tracing)
- ロバー操作開発のための仮想(バーチャル)環境の使用
  - 数日間のミッションリハーサル

# ありがとうございました

Special thanks to:

Open Robotics & NASA ames for inspiration:

<https://www.osrfoundation.org/gazebo-renders-the-moon/>

意見・感想

Louis Burtz: [l-burtz@ispace-inc.com](mailto:l-burtz@ispace-inc.com)

Fabian Dubois: [f-dubois@ispace-inc.com](mailto:f-dubois@ispace-inc.com) / [fabian@datamaplab.com](mailto:fabian@datamaplab.com)